

Reihe: Telekommunikation @ Mediendienste · Band 16

Herausgegeben von Prof. Dr. Dr. h. c. Norbert Szyperski, Köln, Prof. Dr. Udo Winand, Kassel, Prof. Dr. Dietrich Seibt, Köln, Prof. Dr. Rainer Kuhlen, Konstanz, Dr. Rudolf Pospischil, Brüssel, Prof. Dr. Claudia Löbbecke, Köln, und Prof. Dr. Christoph Zacharias, Köln

PD Dr.-Ing. habil. Martin Engelen
Prof. Dr.-Ing. habil. Klaus Meißner (Hrsg.)

Virtuelle Organisation und Neue Medien 2004

Workshop GeNeMe2004
Gemeinschaften in Neuen Medien

TU Dresden, 7. und 8. Oktober 2004



Bibliographische Information der Deutschen Bibliothek

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliographie; detaillierte bibliographische Daten sind im Internet über <<http://dnb.ddb.de>> abrufbar.

ISBN 3-89936-272-1
1. Auflage September 2004

© JOSEF EUL VERLAG GmbH, Lohmar – Köln, 2004
Alle Rechte vorbehalten

Printed in Germany
Druck: RSP Köln

JOSEF EUL VERLAG GmbH
Brandsberg 6
53797 Lohmar
Tel.: 0 22 05 / 90 10 6-6
Fax: 0 22 05 / 90 10 6-88
E-Mail: info@eul-verlag.de
<http://www.eul-verlag.de>

Bei der Herstellung unserer Bücher möchten wir die Umwelt schonen. Dieses Buch ist daher auf säurefreiem, 100% chlorfrei gebleichtem, alterungsbeständigem Papier nach DIN 6738 gedruckt.



Technische Universität Dresden - Fakultät Informatik
Privat-Dozentur Angewandte Informatik, Professur Multimediatechnik

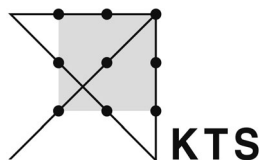
PD Dr.-Ing. habil. Martin Engelen
Prof. Dr.-Ing. habil. Klaus Meißner
(Hrsg.)



an der
Fakultät Informatik der Technischen Universität Dresden

in Zusammenarbeit mit der
Gesellschaft für Informatik e.V.
GI-Regionalgruppe Dresden

gefördert von der Klaus Tschira Stiftung



KLAUS TSCHIRA STIFTUNG
GEMEINNÜTZIGE GMBH

am 07. und 08. Oktober 2004 in Dresden
www.geneme.pdai.de
geneme@pdai.de

D.8 Wissensaustausch in Open Source Projekten

Manfred Langen, Thorbjørn Hansen

Siemens AG, Corporate Technology, Fachzentrum Wissensmanagement

1. Einleitung

Weitgehend unbeobachtet von der Disziplin Wissensmanagement haben sich seit vielen Jahren Open Source Softwareentwickler erfolgreich im Wissensaustausch geübt. Ohne direkten Auftrag- oder Geldgeber arbeiten sie über lange Zeiträume und häufig sehr erfolgreich als virtuelles Team zusammen. Die dabei wirksamen Mechanismen sind von Unternehmen bislang wenig untersucht oder beachtet worden. Können sie davon profitieren, indem sie die Motivationsgrundlage von Open Source Projekten auf sich „übertragen“?

2. Open Source als Phänomen aus Sicht des Wissensmanagements

Open Source (OS) Software ist nicht erst seit Linux ein Phänomen, dem auch in industriellen Unternehmen Beachtung geschenkt wird. Jedoch bezieht sich diese Beachtung bislang überwiegend auf die Frage, ob eine bestimmte Open Source Software eingesetzt werden soll.

Aus der Sicht des Wissensmanagement ist aber vor allem ein anderer Aspekt interessant, nämlich wie Motivation und Organisation in solchen Projekten funktionieren. In Hinblick auf die Zukunftsvisionen von „virtuellen Unternehmen“ könnte sich die Open Source Szene als Fundgrube für neue Methoden und Technologien herausstellen. Gerade die Mechanismen für den Wissensaustausch verdienen besondere Beachtung, da hier in Unternehmen oft Defizite in der Kultur der Zusammenarbeit festgestellt werden. Versuche, diese Defizite durch Incentive-Systeme zu kompensieren, scheinen sich langfristig eher kontraproduktiv auf den Wissensaustausch auszuwirken. Daher werden Alternativen gesucht, die weniger den unmittelbaren (individuellen) geldwerten Vorteil in den Vordergrund stellen.

Dabei ist das Phänomen Open Source nicht gerade neu: Seit 20 Jahren gibt es Open Source Projekte, die zum Teil qualitativ hochwertige Ergebnisse erzielt haben, wie z. B. Linux, GNU, Apache und Mozilla. Und das, obwohl die klassischen Rahmenbedingungen von Softwareprojekten fehlen:

- Es existiert kein Budget.
- Es existiert kein Auftraggeber.
- Es gibt keine Zielvereinbarungen auf feste Termine.
- Projektstrukturen entstehen adhoc und lösen sich ggf. wieder auf .
- Die Softwareersteller erhalten kein Entgelt für ihren Beitrag.

Stattdessen beobachtet man:

- International verteilte Arbeitssysteme mit Selbstorganisation und informeller Planung.
- Selbstbestimmung bei den Zielen.
- Kostenfrei verfügbare Collaborations-Plattformen im Internet.
- Freiwillige, gegenseitige Hilfe.
- Software in Produktqualität zum kostenlosen Download.

Offenbar gibt es explizite und implizite Faktoren, durch die alle Beteiligte kooperativ zusammenwirken, sich permanent selbst organisieren und auf ein gemeinsames Ziel hin ausrichten.

Auffallend ist, dass Entwickler mit einem gewissen Stolz ihr Ergebnis allen kostenlos zur Verfügung stellen. Dabei zeigt sich gleichzeitig ein Qualitätsbewusstsein für das eigene Produkt, das man in manchen Softwarefirmen nur im Marketingprospekt findet. Eine entscheidende Größe für das Funktionieren von Open Source scheint ein System von kollektiven Werten, Denkmustern und Verhaltensweisen zu sein (von einigen Autoren als „Hackerkultur“ bezeichnet), in der diese Projekte entstehen. Diese Kultur steht für ein Paradoxon der Softwareproduktivität: Effizienz wird dadurch erreicht, indem man darauf verzichtet, Effizienz explizit einzufordern.

3. Open Source Entwicklung als Modell für Communities und virtuelle Unternehmen?

Die Verteiltheit heutiger Organisationen hat dazu geführt, dass Communities und Expertennetzwerke als Mittel zum Wissensaustausch an Bedeutung gewonnen haben. Während die einen daran festhalten, dass Communities als freiwillige Interessensgemeinschaft zu betrachten sind, wünschen sich andere wohldefinierte Prozesse und Strukturen für Communities. Innerhalb eines Unternehmens ist wohl der erste Aspekt wichtiger, da der zweite ja bereits durch die Aufbauorganisation abgedeckt wird. Betrachtet man jedoch Communities als Basis für ein virtuelles Unternehmen [Langen, 2000], so müssen beide Aspekte gleichermaßen erfüllt werden. Beispiele für virtuelle Unternehmen sind heute so rar, dass ein Blick auf die Open Source Entwicklung mit ihrer 20-jährigen Erfahrung mehr Aufschlüsse darüber bietet, wie virtuelle Unternehmen gestaltet werden können.

3.1 OS Software Entwicklung

Die wohl bekannteste Plattform für Open Source Projekte ist unter der Adresse sourceforge.net im Internet zu finden. Ein Blick auf den Server von Sourceforge kündigt derzeit mehr als 80.000 Open Source Projekte an, die alleine dort beheimatet sind. Natürlich sind darunter auch viele recht kleine Projekte oder solche, die über die Planungs-

phase nie hinauskommen. Die Fülle zeigt aber deutlich, dass sich OS Software nicht auf Linux und Mozilla beschränkt. Auch weniger bekannte aber nicht minder interessante Projekte wie z. B. die ERP-Software Compiere finden sich dort. Das Besondere ist, dass nicht nur der aktuelle Stand dort zum Download bereitgestellt wird – vielmehr befindet sich die gesamte Entwicklungsumgebung für das verteilte Projektmanagement im Internet.

Somit liegt nicht nur die Software offen, sondern auch der gesamte Entwicklungsprozess inklusive aller Diskussionen ist im Internet abgelegt und im Bedarfsfall recherchierbar. Durch dieses Vielaugenprinzip finden eine ständige Qualitätsüberwachung und eine schnelle Fehlerbeseitigung statt, was der Open Source Software Robustheit und Sicherheit verleiht.

In großen Projekten wie Linux oder Apache besteht die Software aus vielen hierarchisch strukturierten Modulen. Innerhalb dieses hierarchischen Baums ist an jedem Blatt ein Autor zu finden, während an allen Knotenpunkten Integratoren zu finden sind. Normalerweise folgt die Besetzung dieser Rollen der Regel des ersten Zurufs (d.h. wer sich als erster freiwillig meldet, hat die Rolle), gelegentlich findet auch ein Wechsel der Zuständigkeit in Folge eines Wettbewerbs statt. Für den Fall, dass keine Einigung erzielt wird, kann ein „forking“ zu zwei parallelen Entwicklungslinien führen. Dies ist i.d.R. als Sanktion gegen die Ursprungslinie gerichtet und daher ein ungern gesehener Ausweg innerhalb einer evolutionären Entwicklung.

Open Source Lizenzbestimmungen gewähren der OS Software einen gewissen „Rechtsschutz“. Die „General Public License (GPL)“ – die populärste OS Lizenz – legt z. B. fest, dass ein Nutzer weder die Software selbst noch Anpassungen derselbigen (Derivate) im Lizenzgeschäft vertreiben kann. Ein kommerzielles Angebot unter Einbeziehung der Software ist jedoch erlaubt und wird von Firmen wie SuSE und RedHat betrieben.

3.2 Community ohne Vertrauen?

In den gängigen Community-Modellen wird immer wieder betont, wie wichtig das gegenseitige Vertrauen für die Lebensfähigkeit einer Community ist. Als wichtiger Hebel werden daher persönliche Treffen gesehen, damit ein Vertrauensverhältnis aufgebaut werden kann. Gerade das ist aber in der Open Source Community nicht gegeben. Warum kann diese Community trotzdem funktionieren?

Ein Blick auf die wissenschaftlichen Erkenntnisse zur Evolution von Kooperation hilft hier weiter [Axelrod, 1984]. Die Spieltheorie hat am Beispiel des Gefangenendilemmas gezeigt, dass Kooperation auch ohne Kenntnis des Mitspielers und ohne Vertrauen aus reinem Eigennutz heraus eine sinnvolle und stabile Strategie sein kann. Die erfolgreichste Strategie TIT FOR TAT beruht alleine auf Reziprozität mit einem gutwilligen

ersten Schritt. Einzige Voraussetzung ist, dass die Interaktion langfristig stattfindet und die Spieler immer wieder aufeinander treffen können. Reziprozität scheint also der Schlüsselfaktor der Kooperation zu sein, wenn man die Mitspieler nicht kennt und auch keine übergeordnete Instanz existiert, die die Unkooperativen bestraft. Auch hier zeigt sich, dass künstliche Anreize durch Incentivesysteme in einer Gruppe von Gleichberechtigten überflüssig sind.

Ein Vertrauen über persönliche Beziehungen ist nicht notwendige Voraussetzung. Wenn man in der OS Community überhaupt von Vertrauen sprechen kann, dann ist es ein generelles Vertrauen in die Prinzipien von OS Software, die auf der Grundlage von Transparenz aufbauen.

Der Einstieg in das OS Umfeld geschieht überwiegend in der Rolle des Nutznießers. Ich kann etwas benutzen, das ich mir einfach nehmen kann und spare damit entweder Geld oder den Aufwand, es selbst zu programmieren. Damit bin ich noch nicht Teil der OS Community, habe ihr aber immerhin meine Aufmerksamkeit und mein Interesse durch einen Download bekundet. Im nächsten Schritt suche ich ein paar Tipps und benutze das dazugehörige Online-Forum. Hier findet dann für viele der Eintritt in die Peripherie der Community statt. Durch regelmäßige Nutzung der Online-Foren findet häufig ein Rollenwechsel statt vom reinen Leser hin zu jemandem, der auch selber Beiträge schreibt. Auf diese Weise wird mit der Zeit aus einem reinen Nutzer ein Beteiligter.

4. Motivationsfaktoren in OS Softwareprojekten

In Bezug auf die Motivationsfaktoren für den Wissensaustausch bietet das Phänomen Open Source einige interessante Erkenntnisse. Wie wir sehen, gibt es eine ausreichende, überwiegend intrinsische Motivation, die eine Kooperation zwischen untereinander fremden Personen, die weltweit verteilt sein können, herbeiführt. Nachfolgend werden 4 Motive erörtert, die sich aus verschiedenen Studien herauskristallisieren.

4.1 Geben und Nehmen

Eine Studie des MIT über das Apache Usenet help system [MIT, 2000] hat ergeben, dass eine Symmetrie im Geben und Nehmen eine entscheidende Größe für die Motivation der Befragten war, in OS Softwareprojekten mitzuarbeiten. Diese Symmetrie muss nicht unmittelbar gegeben sein, sondern wird längerfristig im Sinne einer Tauschökonomie betrachtet. So bekommt derjenige, der Software schreibt, Feedback über die Funktionstüchtigkeit bzw. korrigierte Codesegmente zur Fehlerbeseitigung. Wer in den Foren einen hilfreichen Beitrag findet, ist häufig motiviert, auch jemand anderem einen hilfreichen Tipp zu geben. Solange ein Nehmen und wieder Zurückgeben bzw. Geben und wieder Zurückbekommen stattfindet, sind die Teilnehmer zufrieden und infolgedessen hat die Lebensfähigkeit des Projektes eine gute Prognose.

Die MIT-Studie bestätigt somit das Motiv einer Reziprozität – ganz im Sinne der TIT FOR TAT Strategie.

4.2 Spaß am Lernen

Eine große Gruppe von OS Interessierten findet sich unter Studenten. Für sie ist die Möglichkeit, Fähigkeiten und Fertigkeiten an einem interessanten Projekt weiterentwickeln zu können, ein starkes Motiv. Offenbar ist es in OS Softwareprojekten so, dass man beim Lernen auch Spaß haben kann. Sicherlich ist es kein Zufall, dass Linus Torvalds sein Buch unter dem Titel „Just for fun“ publiziert hat [Torvalds, 2002]. Betrachtet man dagegen die schleppenden Bemühungen, das Thema e-Learning voranzubringen, so hat sich im OS Umfeld bereits eine virtuelle Lernplattform etabliert, ohne dass jemand sie gezielt als solche geplant hat.

4.3 Reputation

Ein weiterer Motor, sich als Entwickler an OS Projekten zu beteiligen, ist, als Autor bekannt zu werden. Da eine Software schneller größere Verbreitung findet, wenn sie kostenlos ist, wiegt für viele das Bekanntwerden den Verzicht auf ein direktes Entgelt auf. Es kann natürlich argumentiert werden, dass die Reputation langfristig zu besseren Karrieremöglichkeiten und damit wieder in Geld umgesetzt werden kann. Diese Art von Berechnung scheint jedoch für viele nicht im Vordergrund zu stehen. Es geht ihnen meist gar nicht um die Anerkennung von einer Firma, in der Kompensation in Form von Geld und Karriere angeboten wird, sondern um Anerkennung unter ihresgleichen, den Experten für Softwareentwicklung. Die Experten erkennen die Kompetenzträger an ihrem Werk – eine Selbstdarstellung in Form von schönen Reden und bunten Folien ist überflüssig.

4.4 Selbstbestimmung

Ein anderes von Entwicklern genanntes Motiv ist, in OS Softwareprojekten das tun zu können, was man wirklich möchte und das freiwillig zu tun. Wenn das Ergebnis dann bei anderen Beachtung findet, ist es Anerkennung dafür, was man gemacht hat und nicht dafür, dass man es so gemacht hat, wie jemand aufgetragen hat. Daraus kann man schließen, dass Eigenständigkeit und Selbstbestimmung der Kreativität in der Softwareentwicklung sehr förderlich sind, während eine definierte Kunden-Lieferanten-Situation eher blockierend wirkt. Im OS Umfeld ist es jederzeit möglich, als „Kunde“ unmittelbar Einfluss auf eine bestehende Software zu nehmen.

4.5 Die „Hackerkultur“

Die genannten Motive verbinden die Akteure in einer OS Community und bilden so eine Basis für deren Kooperationsverhalten. Eine gemeinsame politische Komponente ist teilweise vorhanden, wurde aber in der Vergangenheit häufig überschätzt: Es besteht eine Abneigung gegen die Restriktionen bei der Nutzung kommerzieller Software durch Lizenzregelungen, die als mehr oder weniger kompliziert mit undurchsichtigen rechtlichen Konsequenzen angesehen werden.

Dies ist aber weniger als „Glaubensmodell“ oder als „Gegenoffensive“ zu Microsoft zu verstehen. Man ist ganz einfach davon überzeugt, dass sich „Software-Monokulturen“ als problematisch erwiesen haben und die damit verbundenen Abhängigkeiten mehr als hinderlich sind. Quellcode soll daher offen zugänglich sein, weil diese Form der Transparenz Voraussetzung für Qualität und schnelleren technischen Fortschritt ist. Die Verfügbarkeit des Codes ermöglicht es, unmittelbar selbst Fehler zu beheben und zusätzlich benötigte Funktionalität einzubauen ohne auf die nächste angekündigte Version warten zu müssen. Luftige Marketingversprechungen haben wenig Chancen in dieser Kultur, da die Karten offen auf den Tisch – in diesem Fall auf den Server – gelegt werden.

5. Konsequenzen für Unternehmen

Mit OS Software ist der Beweis erbracht, dass die Nutzung geistiger Ressourcen ohne nennenswerten Geldmitteleinsatz möglich ist – sogar in großen Projekten. Allerdings gelingt dies in einem Kontext, der von dem eines Unternehmens stark abweicht.

Für ein Unternehmen stellen sich aufgrund der beobachteten Phänomene viele Fragen:

- Ist Wissen überhaupt noch ein Geschäft, wenn schon Software (vielfach als typisches Wissensprodukt angeführt) frei verfügbar ist?
- Ist Software gar kein Produkt, sondern eher eine Dienstleistung?
- Haben wir in Unternehmen organisationspsychologisch betrachtet die falschen Strukturen für Softwareentwicklung?
- Lässt sich die Bereitschaft zum Wissensaustausch, so wie sie in OS Communities gegeben ist, auf die Zusammenarbeit innerhalb einer Firma übertragen?
- Ist eine mangelnde Kooperationsbereitschaft Folge der hierarchischen Aufbauorganisation, die z. B. eine Konkurrenz um die Besetzung der nächst höher bezahlten Stelle induziert?
- Kann ein klassisches Unternehmen sich überhaupt in eine virtuelle Organisation transformieren und wenn ja, wie?

Erste Antworten finden wir bei einem der bekanntesten Managementtheoretiker: Peter Drucker schrieb 1998 in „Management’s new paradigm“: „Es ist nicht Aufgabe der Manager, anderen zu sagen, was zu tun ist, sondern sie als Freiwillige zu sehen, die sich auf ein gemeinsames Ziel hin selbstverpflichtet haben und an der Steuerung ebenso be-

teilt sein wollen wie am Erfolg.“ Drucker war es auch, der 1993 als erster den Begriff der Wissensgesellschaft prägte und die steigende Bedeutung von Wissensarbeitern betonte. Offenbar ist die Kultur der Technologieexperten anders als die der Ertragsoptimierung. Die Skalierbarkeit des Profits durch Lizenzvertrieb ist im Modell der OS Software eingeschränkt oder ganz unterbunden. Das Dienstleistungsgeschäft durch Experten bleibt jedoch bestehen.

Ein weiteres Spannungsfeld entsteht dadurch, dass möglicherweise die Loyalität der Wissensarbeiter zu ihrer Profession größer ist als zum aktuellen Arbeitgeber. Wie kann also ein Unternehmen eine ausreichende Bindung seiner Wissensarbeiter erreichen, eine Frage, die sich für virtuelle Organisationen noch viel vehementer darstellt als ohnehin schon für die heutigen Firmen?

Sicher scheint zu sein, dass jeder, der etwas über das virtuelle Unternehmen lernen will, sich mehr an den Prinzipien von OS orientieren sollte, um aus dieser Erfahrung heraus weitere Schlüsse zu ziehen.

6. Unser Lernexperiment

Die Corporate Technology der Siemens AG verfolgt schon seit langer Zeit die wichtigen Trends für die Zukunft verteilter Unternehmen. Im Fachzentrum für Wissensmanagement betrachten wir dabei besonders die Methoden und Technologien zur Zusammenarbeit. Die vorgenannten Überlegungen haben uns veranlasst, das Phänomen der OS Software nicht nur zu beobachten, sondern Erfahrungen mit einem eigenen Projekt zu machen, um aus „erster Hand“ zu lernen. Hierfür haben wir eine intern entwickelte Portalsoftware ausgesucht.

Die Portalsoftware des Siemens Employee Portals ist seit 1. Oktober 2003 unter dem Namen Siemens Intranet Portal Framework (SIPF) auf sourceforge.net im Internet verfügbar (siehe <http://sipf.sourceforge.net>). Das Siemens Intranet Portal Framework wurde 1999 mit dem Projektziel begonnen, das Firmenintranet mit seinen mehr als 2500 Servern und mehreren Millionen Seiten als größte unternehmensweit verfügbare Informationsquelle zu strukturieren, standardisieren und konsolidieren. Aufgabe des Siemens Employee Portals ist es, durch eine bessere Unterstützung der firmenweiten Informationsflüsse einen Kondensationskeim für die Produktivitätsvorteile von e-Business zu bilden. Seit Oktober 2001 ist das Employee Portal im operativen Betrieb – mit mehr als 100.000 Benutzern. SIPF bietet einen strukturierten Zugang zu Informationen, eine nahtlose Integration von Applikationen und Diensten sowie Hilfe durch Teil-Automatisierung der zugrunde liegenden Prozesse. Hierfür können Applikationen und Dienste als Bausteine eingebunden und ein zusätzlicher Mehrwert durch applikationsübergreifende Interaktion erzielt werden.

Durch die Veröffentlichung soll die Verantwortlichkeit für diese Software Schritt für Schritt in die Hände einer OS-Community übergehen. Mit der im Laufe dieses Prozesses gewonnenen Erfahrung erwarten wir uns die Beantwortung der oben gestellten Fragen sowie den Aufbau einer generellen Consulting-Expertise zum Umgang mit dem Thema Open Source.

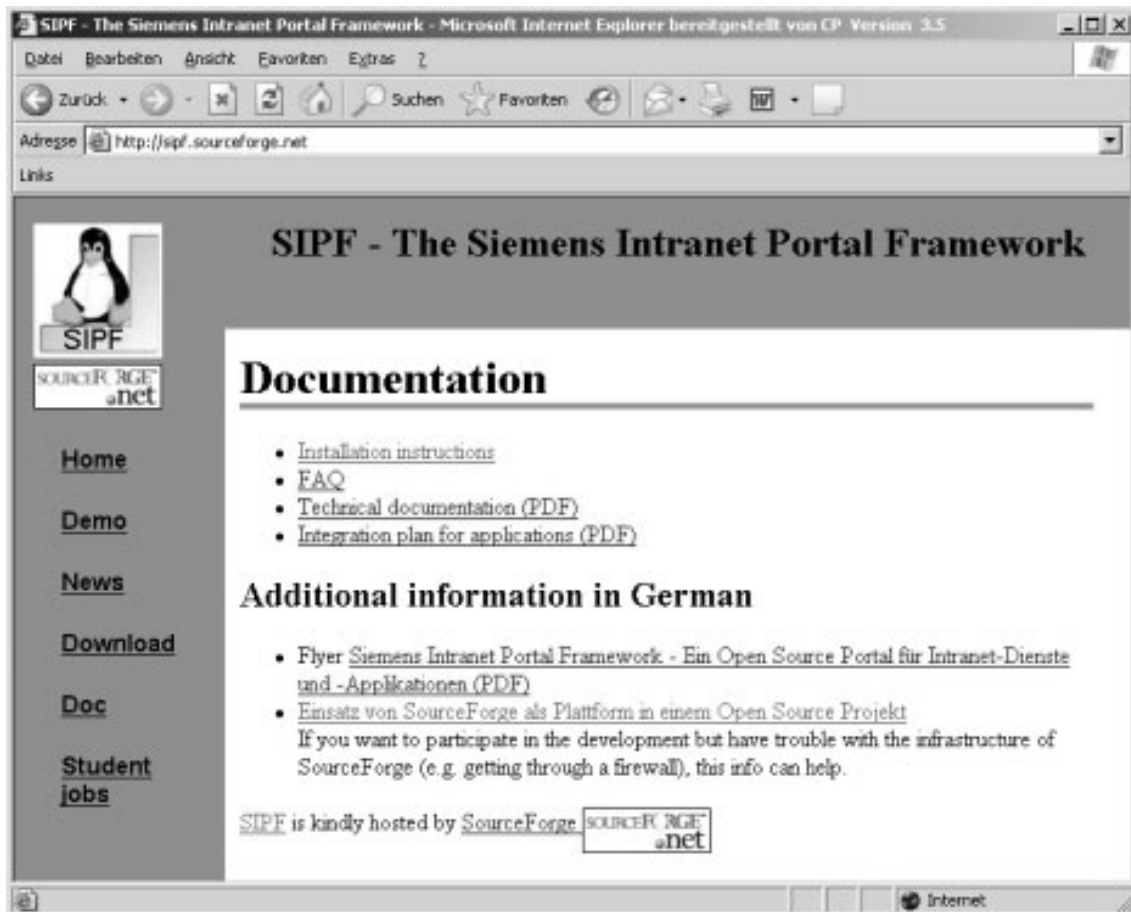


Abbildung 1: Projekt Site auf Sourceforge

7. Erste Erfahrungen

Die allerersten Erfahrungen waren eher technischer Natur. So wurde zunächst eine Softwarevariante erstellt, die ausschließlich auf der Basis von OS Software aufsetzen kann: Linux, Apache, Tomcat und Postgres. Neben der Vereinfachung der Installation machte auch die haus eigene Firewall zunächst ein paar Probleme. Erheblich schwieriger ist jedoch, das Projekt bekannt zu machen und unter der Vielzahl von Projekten aufzufallen. Auf diese Weise erfährt man, was es heißt, sich in einer Aufmerksamkeitsökonomie zu bewegen, wo auf einer Plattform 80.000 Projekte beachtet werden wollen. Eine Untersuchung der Ranking Mechanismen, die es erlauben, im Activity Level der

Projekte aufzusteigen, führte zu einer überraschenden Erkenntnis: eine (fast) fertige Lösung schneidet gegenüber einer entstehenden mit vielen Bugs und Änderungswünschen im Ranking schlechter ab und bekommt dadurch weniger Aufmerksamkeit. Diese Konsequenz erscheint zunächst widersinnig, ist aber aus dem Selbstverständnis einer Entwicklungsplattform heraus nachvollziehbar.

Desweiteren zeigt sich, dass eine quasi fertige Software mit der Komplexität eines Portals eine größere Hürde für die Einarbeitung darstellt. Dagegen haben viele bottom-up Projekte mit stetigem, evolutionärem Wachstum weniger Barrieren für den Einzelinteressenten. Daher sind es derzeit eher mittelgroße Firmen, die sich für unseren Ansatz interessieren.

Falls Sie Interesse bekommen haben, dann nehmen Sie einfach Kontakt mit uns auf oder besuchen Sie die Internetseiten des Projekts auf <http://sourceforge.net/projects/sipf/>

8. Literatur

Raymond, Eric S.; The Cathedral & the Bazaar; O'Reilly 2001; ISBN 0596001088

Moody, Glyn; The Rebel Code; Perseus Publishing 2001; ISBN 0713995203

Axelrod, Robert; The Evolution of Cooperation; Basic Books 1984; ISBN: 0465021220

Drucker, Peter; Management's new paradigm; Forbes October 1998, S.152-177

MIT Sloan School of Management; How open source software works: free user-to-user assistance; Working Paper 4117-00 May 2000

Langen, Manfred; e-Business Communities – die soziotechnische Basis-Infrastruktur für virtuelle Unternehmen; KnowTech 2000 in Leipzig

Torvalds, Linus; Just for fun; HarperBusiness 2002; ISBN: 0066620732

